# Emulator with Switching Network Connections

Terry Lee Goode

## BACKGROUND OF THE INVENTION

### 1. Field of the invention

The present invention relates to hardware emulation of integrated circuits. In particular, the present invention relates to an emulator architecture implemented using serialized interconnections between programmable logic devices.

### 2. Discussion of the Related Art

Hardware emulation of integrated circuits is a method extensively used in the design process of complex logic circuits. A hardware emulator provides a substrate for implementing a complex logic circuit. Typically, the emulator includes as building blocks of the substrate one or more interconnected circuit boards. In each circuit board, a number of interconnected programmable logic devices are provided for implementing selected portions of the complex logic circuit. Field programmable gate arrays ("FPGAs") are the most common programmable logic devices ("PLDs") found in a hardware emulator. The hardware emulator is controlled by software running on a host processor, such as an engineering workstation.

To achieve hardware emulation of the complex logic circuit, a designer provides a description of the logic circuit in a hardware description language. In a modern hardware emulator, the description of the logic circuit can be provided as a behavior level description, a register transfer level (RTL) description (e.g., a verilog or VHDL description), or a logic gate level netlist. If a behavior or RTL level description is used, a logic gate level description for the circuit is synthesized. From the logic

gate level description, the logic circuit is partitioned. Each partition is assigned to a PLD for realization. Signals between partitions implemented on different PLDs are transmitted over the pins of the PLDs. The physical signal paths for routing such signals depend on the how the PLDs are laid out and interconnected on each circuit board, and how the circuit boards are interconnected. In some emulators, partial cross bar switches are provided to route signals between PLDs. In other emulators, each PLD is directly connected to a fixed number of other PLDs directly, and another number of PLDs indirectly, according to a pre-determined interconnection configuration. Often, signals between partitions implemented on different PLDs are routed through one or more intermediate PLDs.

In a large logic circuit implemented over multiple PLDs, the pins of the PLDs become a scarce resource that must be carefully allocated to avoid excessive signal delays and to maintain high resource utilization in the PLDs. Complex algorithms for placement of partitions and signal routing are devised to achieve these goals. However, a structural organization of the PLDs and their associated interconnection circuits that simplifies circuit placement and signal routing is desired.

## SUMMARY OF THE INVENTION

According to the present invention an emulator synthesizes circuit partitions from a user circuit. The circuit partitions are configured into programmable logic devices of one or more emulator circuit boards. According one embodiment of the present invention, the output signals of one circuit partition designated for another circuit partition are provided as output signals of a first programmable logic device, and at the other circuit partition, which is implemented in a second programmable logic device, the input signals are received into the circuit partition from input signals of the second

programmable logic device. The output signals of the first programmable logic device are serialialized to provide a serialized data stream, which is received into a cross point switch. The cross point switch routes the serialized data stream from one of its input/output pins onto another one of input/output pins. The data stream is then deserialized as the input signals of the second programmable logic device. The cross point switch may reside on the same circuit board as one or more of the programmable logic devices, but it may also reside on a separate circuit board.

In one embodiment, both a serialization circuit and a deserialization circuit are provided for serializing the output signals and deserializing the input signals of a circuit partition in the programmable logic device.

In one embodiment, a serializer/deserializer integrated circuit provides for each programmable logic devices the serializing and deserializing of its input and output signals. The serializer/deserializer integrated circuits may reside on the same or different circuit board as the first and second programmable logic devices, or the cross point switch. In another embodiment, more than one serializer/deserializer can be provided to support the operation of each programmable logic device.

In one embodiment, the input and output signals of each programmable logic circuit are provided on pins used for implementing virtual interconnection between circuit partitions of the user circuit. In another embodiment, a dedicated pin is provided to each input or output signal.

The cross point switch can be configured for either static or dynamically scheduled operations.

A system for emulation according to the present invention includes, on one or more circuit boards, (a) a first programmable logic device and a second programmable logic device, the first programmable logic device having

pins configurable for providing output signals and the second programmable logic device having pins configurable for receiving input signals; (b) a serializer receiving the output signals from the first programmable logic device and providing a serialized data stream; (c) a cross point switch receiving the serialized data stream at a first input/output pin of the cross point switch and routing the data stream onto a second input/output pin of the cross point switch; and (d) a deserializer receiving the data stream from the cross point switch and deserializing the data stream onto the pins of the second programmable logic device.

The present invention is better understood upon consideration of the detailed description below and the accompanying drawings.

## BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows schematically system 100 for emulating a complex logic circuit, including serializer/deserializers 110 and 114, and cross point switch 112, in accordance with one embodiment of the present invention.

Figure 2 shows schematically system 200 for emulating a complex logic circuit, in which serializer/deserializers 210 and 214 are integrated into programmable logic devices 201 and 202, respectively.

In Figures 1 and 2, to facilitate correspondence between the figures and to simplify the detailed description below, like elements are assigned like reference numerals.

## DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention provides a method and an organization of a hardware emulator that simplify circuit placement and signal routing of the emulated circuit. Figure 1 shows schematically system 100 for emulating a complex logic circuit, in accordance with one embodiment of

the present invention. System 100 includes a number of PLDs (e.g., FPGAs), represented in Figure 1 by PLDs 101 and 102. During operation, a user circuit is partitioned and configured in the PLDs, such as represented in Figure 1 by circuit partitions 103a and 103b. In this embodiment of the present invention, signals received into and transmitted out of each PLD are provided using virtual interconnections, such as described in U.S. Patents 5,854,752 and 5,761,484. In a virtual interconnection, user output signals from circuit partitions designated for circuit partitions in another PLD are time-multiplexed by multiplexers onto input/output (I/O) pins of the originating PLD. For example, in Figure 1, user output signals 104a from circuit partition 103a in PLD 101 designated for circuit partition 104b are time-multiplexed by multiplexer 105a onto I/O pins 107a of PLD 101. A PLD receiving such time-multiplexed signals at its I/O pins demultiplexes the received signals as user input signals into the destination circuit partition. For example, in Figure 1, time-multiplexed signals received at PLD 101's input and output pins 107a are demultiplexed by demultiplexer 106a as user signals 108a into circuit partition 103a. A state machine according to a "virtual clock" controls the timing of the multiplexing and demultiplexing operations. The multiplexers, demultiplexers and state machines are typically synthesized and configured on the PLDs. Various aspects relating to virtual interconnections are described in U.S. Patents 6,104,210, 6,061,511, 5,659,716, 5,649,176, 5,850,537, 5,847,578 and 5,802,348. The disclosures of these U.S. Patents are hereby incorporated by reference as background information regarding virtual interconnections.

In existing hardware emulators implementing virtual interconnections, such as the VStation products from IKOS Systems, San Jose, California, the pins of the PLDs in an emulator board are interconnected by fixed conductors on the emulator board. Various configurations of PLDs and fixed

conductors can be found in the U.S. Patents incorporated by reference above. In accordance with the present invention, however, rather than using fixed conductors, the embodiment of the present invention shown in Figure 1 serializes the virtual interconnection output signals on I/O pins 107a of PLD 101 that are designated for PLD 102 onto conductors 111, which are coupled to I/O pins of cross point switch 112. Cross point switch 112 routes the signals received on conductors 111 in their serialized form onto conductors 115, which are then deserialized onto I/O pins 107b of PLD 102. Signals received on I/O pins of PLD 102 are demultiplexed onto user input signals 108b of circuit partition 103b in the conventional manner for virtual interconnections. Likewise, the virtual interconnection output signals on I/O pins 107b of PLD 102 that are designated for PLD 101 are serialized onto conductors 115 to be routed by cross point switch 112 onto conductors 111. The routed signals are then deserialized onto I/O pins 107a of PLD 101. Cross point switch 112 is connected to a number of PLDs. If necessary, additional cross point switches can be provided such that signals from any PLD in the hardware emulator can be routed to any other PLD in the hardware emulator. PLDs 101 and 102, serializer/deserializaer 110 and 114 and cross point switch 112 can reside on the same or different circuit boards. The operation of cross point switch 112 can be controlled by a state machine (represented in Figure 1 by state machine 113). Further, the configuration of cross point switch 112 can be "static" (i.e., fixed at compile time for each netlist implemented on the hardware emulator), or "dynamic" (i.e., provided according to a state machine that operates a compiled scheduling process). The techniques for scheduling virtual interconnections can be used in dynamic scheduling of signals routed through cross point switch 112. In some circumstances, the control structure is significantly simplified when each serialized data stream includes only signals originating from only a single programmable logic device and designated only for one

destination programmable logic device. In other circumstances, data from different PLDs can be combined into the serialized data stream.

In the embodiment of the present invention shown in Figure 1, serialization and deserialization (represented by serializer/deserializer ("serdes") 110) can be performed by a serial backplane device such as the S2004 integrated circuit available from Applied Micro Circuits Corporation, San Diego, California. The S2004 integrated circuit includes a phase-locked loop circuit that generates a high speed clock from an input clock signal, and uses the high speed clock to serialize input data. In one implementation, the high speed clock is embedded in the data stream to be recovered upon deserialization. Further, cross point switch 112 can be provided by a cross point switch integrated circuit, such as the S2016 or the S2025 from Applied Micro Circuits Corporation, San Diego, California.

Alternatively, serdes 110 can be integrated into PLD 101 and 102 and provided as a configurable resource, which can be allocated and configured by software at the same time the user circuit in the PLDs is configured. Such an integrated circuit would integrate, for example, a conventional FPGA (e.g., an FPGA circuit from Xilinx, Inc.) with a serializer/deserializer circuit, such as that licensed by RocketChip, Inc. For example, Figure 2 shows schematically system 200 for emulating a complex logic circuit, in which serializer/deserializers 210 and 214 are integrated into programmable logic devices 201 and 202, respectively. (In Figures 1 and 2, to facilitate correspondence between the figures and to simplify the detailed description below, like elements are assigned like reference numerals.)

The above detailed description is provided to illustrate the specific embodiments of the present invention and is not intended to be limiting. Numerous modification

and variations within the scope of the present invention are possible. For example, even though the embodiment of the present invention in Figure 1 above shows that a serial data stream grouping output data from virtual interconnections, virtual interconnections are not necessary for practicing the present invention. The configurer can provide each input/output signal of a circuit partition a dedicated signal path. The present invention is set forth in the following claims.